

# Point Operation

## Histogram Modification

김성영교수

금오공과대학교

컴퓨터공학과

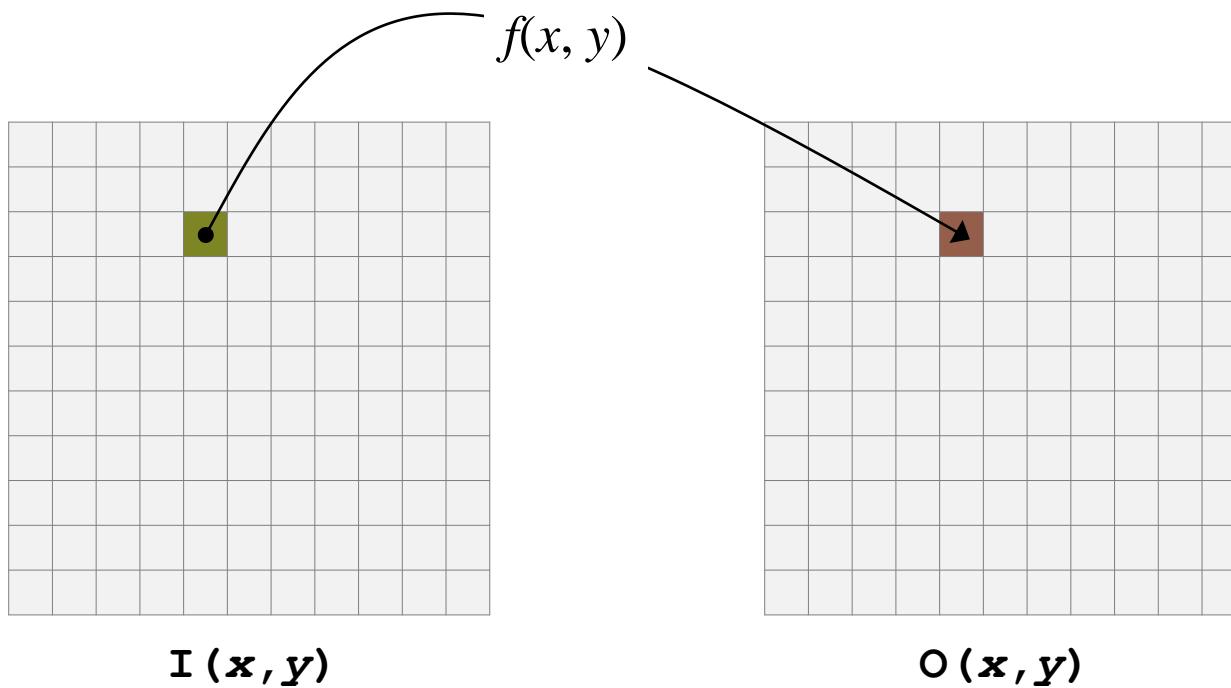
# 학습 내용

---

- POINT OPERATION 개요
- ARITHMETIC OPERATION
- GRayscale TRANSFORMATIONS
- PROCESSING FOR COLOR IMAGES

# POINT OPERATION 개요

- Each pixel value is replaced with a new value obtained from the old one



$I = O$ : in-place transformation

# TECHNIQUES

ARITHMETIC OPERATION

GRAYSCALE TRANSFORMATION

HISTOGRAM MODIFICATION

# OBJECTIVE

Improving image ***contrast*** and ***brightness***

**Image contrast:** a measure of the distribution and range of the gray levels

the difference between the brightness and darkest pixel values, and  
how the intermediate values are arranged

**Image brightness:** the overall average or mean pixel value in the image

## Section 01 화소 점 처리의 개념

### 화소 점 처리

- 원 화소의 값이나 위치를 바탕으로 단일 화소 값을 변경하는 기술
- 다른 화소의 영향을 받지 않고 단순히 화소 점의 값만 변경하므로 포인트 처리(**Point Processing**)라고도 함.
- 산술연산, 논리연산, 반전, 광도 보정, 히스토그램 평활화, 명암 대비 스트레칭 등의 기법이 있음.
- 디지털 영상의 산술연산은 디지털 영상의 각 화소 값에서 임의의 상수 값으로 덧셈, 뺄셈, 곱셈, 나눗셈을 수행하는 것
- 그레이 레벨 영상에서 화소 값이 작으면 영상이 어둡고, 화소의 값이 크면 밝음.

# 산술연산과 논리연산

## 산술연산

- 밝기 조정과 관련된 작업 수행



[그림 4-2] 디지털 영상의 밝기 변화

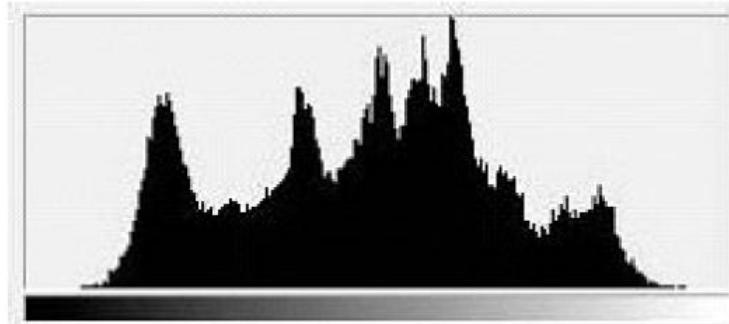
## 논리연산

- 참과 거짓을 판별하는 연산
- 화소의 상수 값에서 AND, OR, XOR, NOT 등의 연산을 수행하여 디지털 영상에서 차폐, 특징 추출, 형태 분석을 함.

# 히스토그램(Histogram)

## ▶ 히스토그램(Histogram)

- 기둥그래프나 기둥 모양 그림이라고도 하며, 관측한 데이터가 분포된 특징을 한눈에 볼 수 있도록 기둥 모양으로 나타낸 것.
- 가로축에는 레벨(Level)을, 세로축에는 각 레벨의 빈도수를 표시함.
- 즉, 가로축은 영상의 밝기(Intensity) 값, 세로축은 가로축의 밝기 값에 대응하는 디지털 영상 내의 화소 수



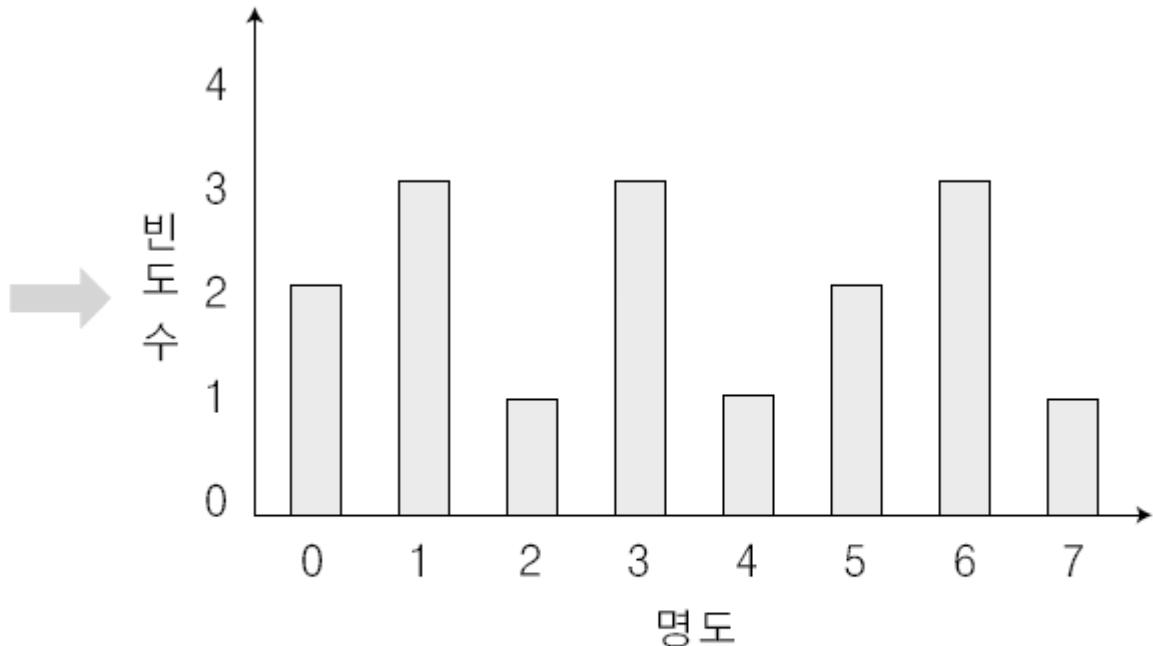
[그림 4-3] 디지털 영상의 밝기 히스토그램

## Section 01 디지털 영상의 히스토그램

### ▣ 디지털 영상의 히스토그램

- 관찰한 데이터의 특징을 한눈에 알아볼 수 있도록 데이터를 막대그래프 모양으로 나타낸 것
- 디지털 영상에 대한 많은 정보를 제공함.

6	6	6	7
4	5	5	3
2	1	1	3
0	0	1	3



(a) 입력 영상

(b) 히스토그램

[그림 5-1] 이상적인 영상의 히스토그램

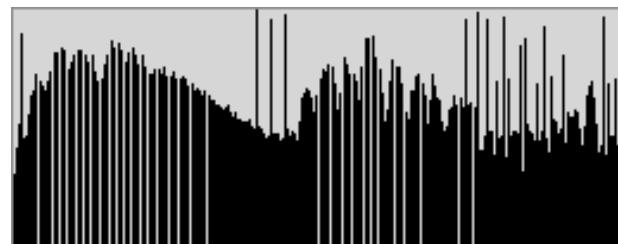
# HISTOGRAM

- A simple datum that gives **the number of pixels that a given value in an image**
- Ex) a 8bit gray-scale image



Bin	Counts	Prob.
0	163	0.005
1	77	0.003
...		
255	1561	0.051

number  
of  
pixels



gray level

## 화소 값의 덧셈연산

- 화소의 밝기 값에 특정한 상수 값을 더해 화소의 밝기 값을 증가시켜 영상을 밝게 하는 처리 기술

화소 +  $\alpha$  : 영상의 밝기 증가 = 밝아짐

- 화소의 값에 임의의 상수를 더할 때 화소의 최대값을 넘기도 함.
- 최대값인 255를 넘는 값은 모두 255로 처리

(화소 값 +  $\alpha$ ) > 255이면, (화소 값 +  $\alpha$ ) = 255

## 화소 값의 뺄셈연산

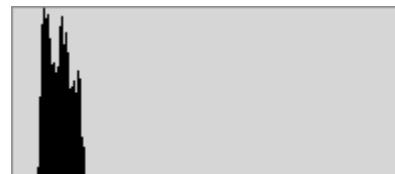
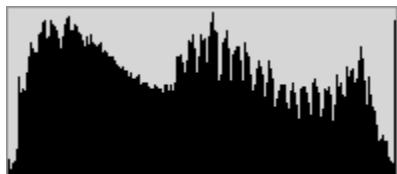
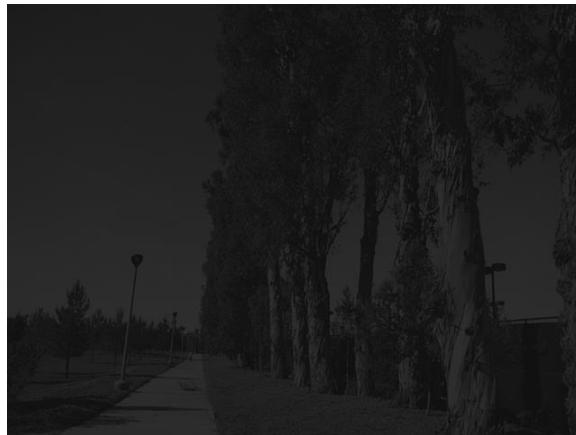
- 화소의 밝기 값에 특정한 상수 값을 빼 화소의 밝기 값을 감소시켜 영상의 밝기를 어둡게 하는 처리 기술

화소  $- \alpha$  : 영상의 밝기 감소 = 어두워짐

- 화소의 값에 임의의 상수를 뺄 때 화소의 최소값 0보다도 작은 음수가 발생 할 수 있음.
- 화소의 최소값인 0보다 작은 음수 값은 모두 0으로 처리

(화소 값  $- \alpha$ ) < 0이면, (화소 값 +  $\alpha$ ) = 0

# CONTRAST & BRIGHTNESS



## 디지털 영상의 산술연산[계속]



(a) 원본 영상



(b) 상수 값 10을 더한 영상



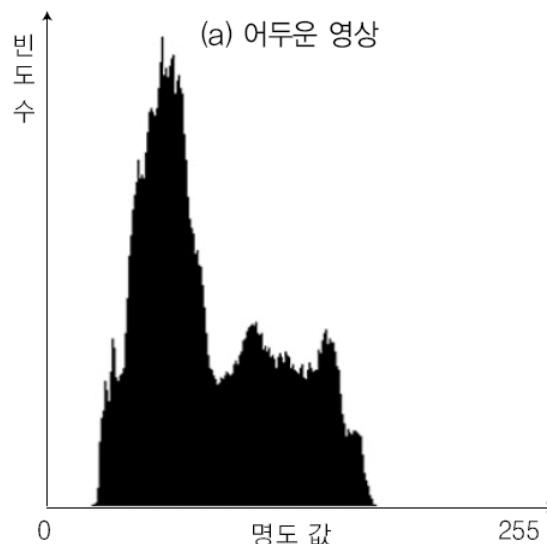
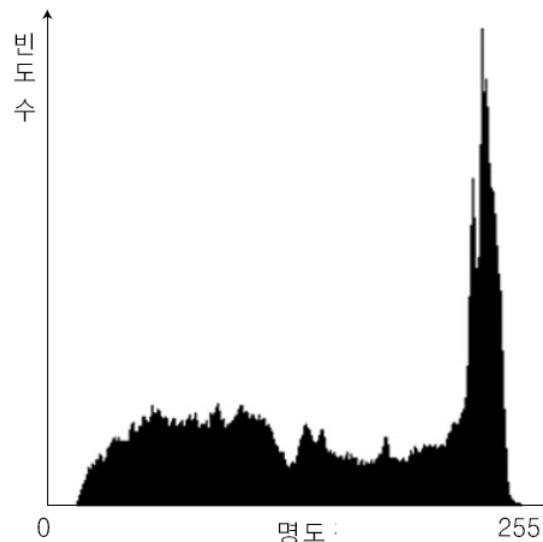
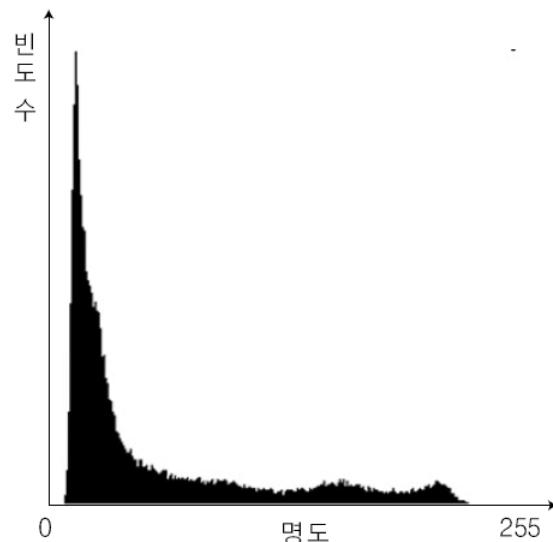
(c) 상수 값 50을 더한 영상



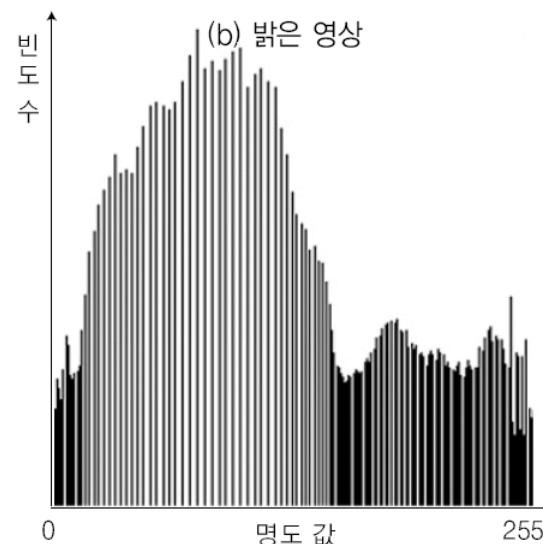
(d) 상수 값 100을 더한 영상

[그림 4-5] 덧셈 상수의 변화에 따른 디지털 영상의 밝기 증가

## 영상의 특성에 따른 히스토그램



(a) 어두운 영상



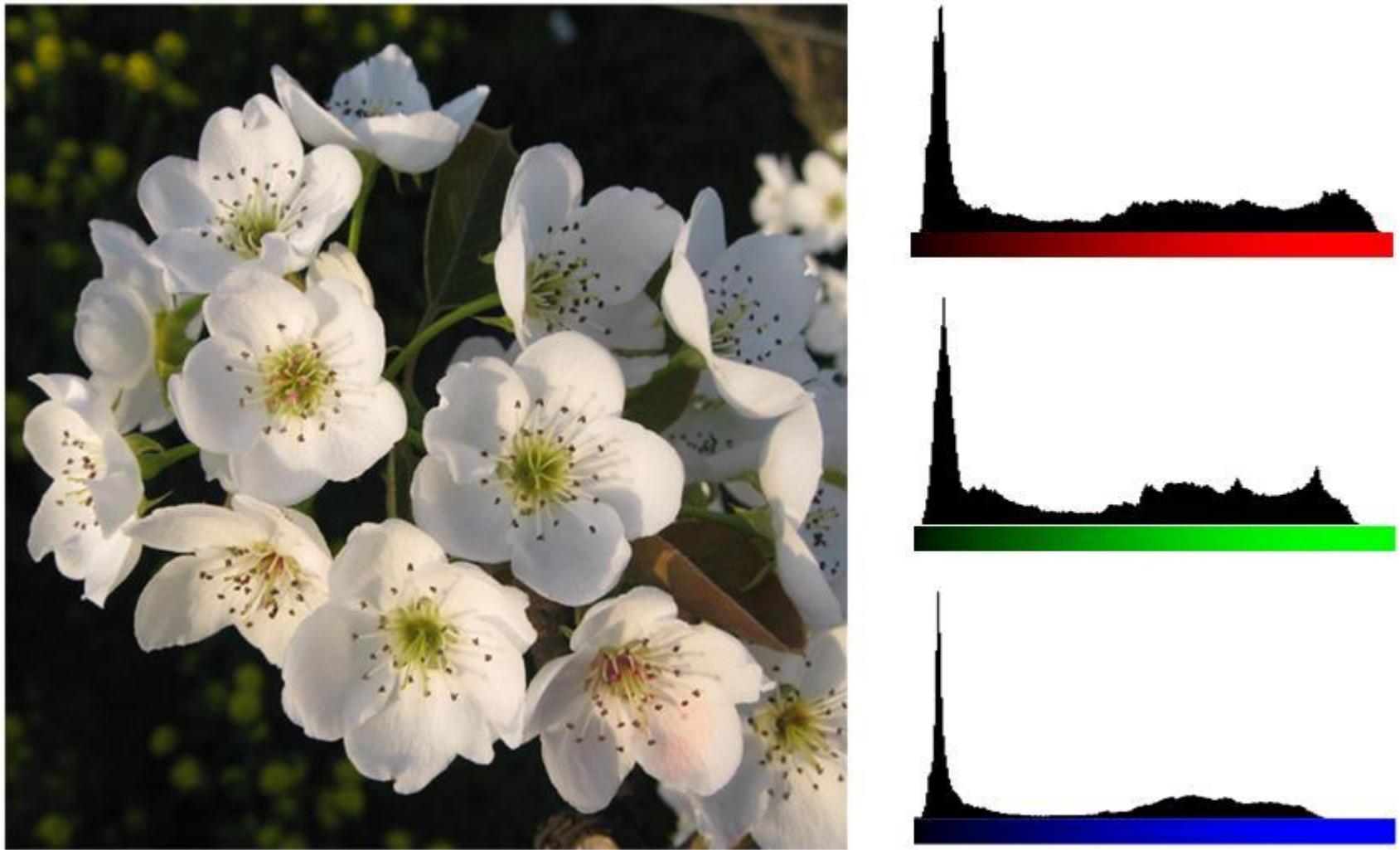
(b) 밝은 영상

(c) 명암 대비가 낮은 영상

(d) 명암 대비가 높은 영상

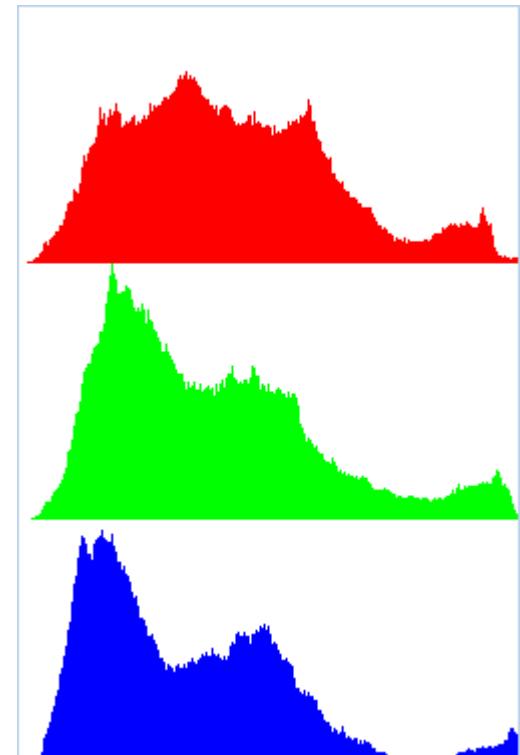
[그림 5-2] 영상의 특성에 따른 히스토그램

## RGB 컬러 영상의 히스토그램

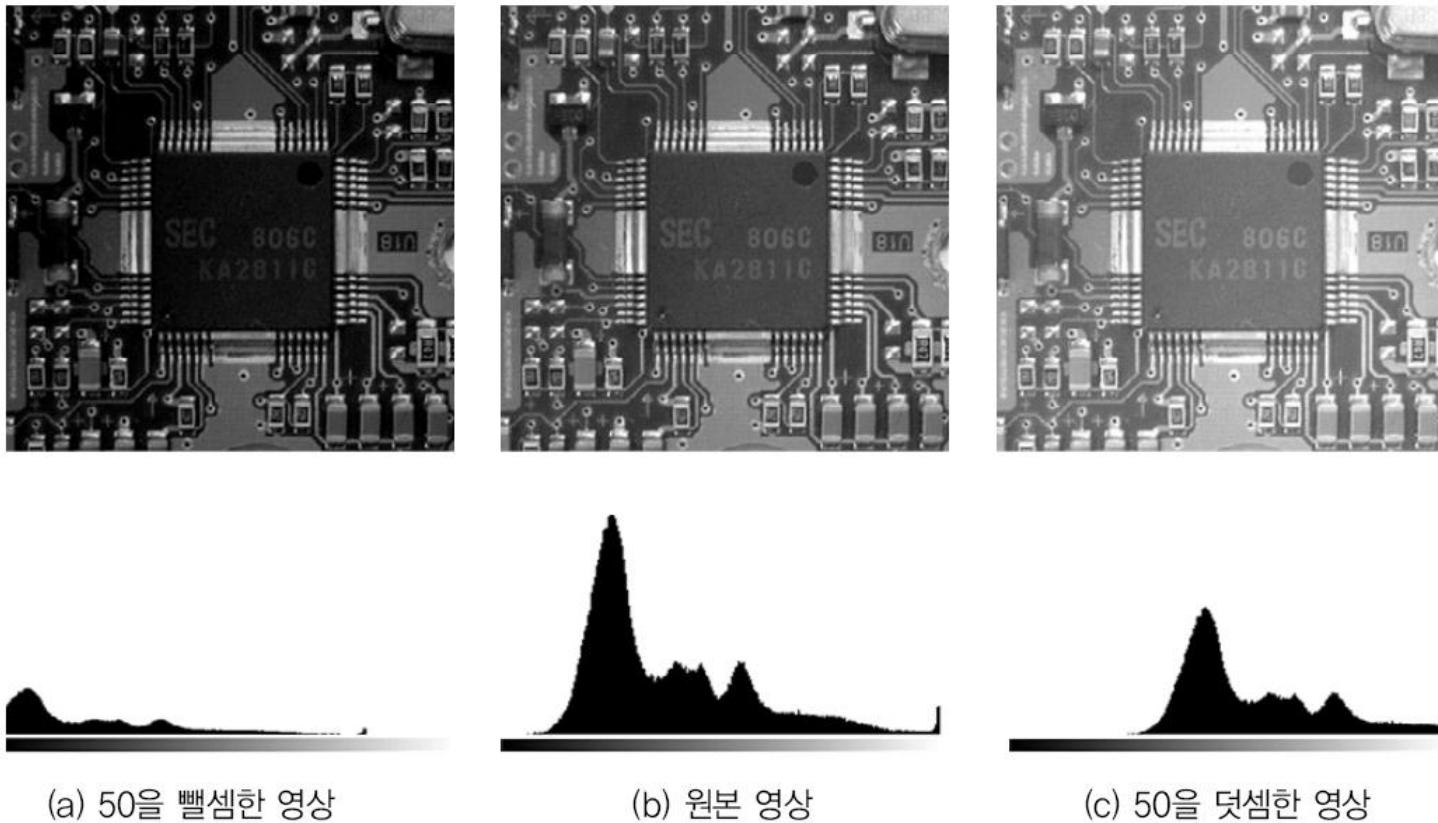


[그림 5-3] RGB 컬러 영상의 히스토그램

# in color images



## Section 02 산술 연산을 이용한 히스토그램 이동



[그림 5-4] 덧셈과 뺏셈연산으로 히스토그램 이동하기

- 덧셈연산: 명도 값을 증가시켜 밝게, 히스토그램의 기둥이 오른쪽으로 이동
- 뺏셈연산: 명도 값을 감소시켜 어둡게, 히스토그램의 기둥이 왼쪽으로 이동

## Section 02 디지털 영상의 산술연산과 논리연산

### 화소의 밝기 값

- 밝기의 단계 수는 화소를 표현하는 양자화 비트 수가 결정
- 그레이 레벨 영상에서는 색은 없고 밝기만 있음.
- 보통, 화소는 밝기를 나타내는데, 주로 양자화 비트 수를 8비트로 표현

### 명암 대비

- 대비(Contrast): 영상 내에 있는 가장 밝은 값과 가장 어두운 값의 차이로, 영상의 품질을 결정하는 중요한 요소임.
- 높은 대비를 보이는 디지털 영상: 어두운 명도와 밝은 명도의 차이가 너무 커서 시각적으로 좀더 명확하게 보임.
- 낮은 대비를 보이는 디지털 영상: 밝기의 차이가 크지 않아 시각적으로 명확하지 못함.

## 화소 값의 곱셈연산

- 화소의 밝기 값에 특정 상수 값을 곱해 전체적으로 화소의 밝기 값이 증가해 더 밝아짐.

화소 \*  $\alpha$  : 영상의 밝기 차이 증가 = 뚜렷해짐

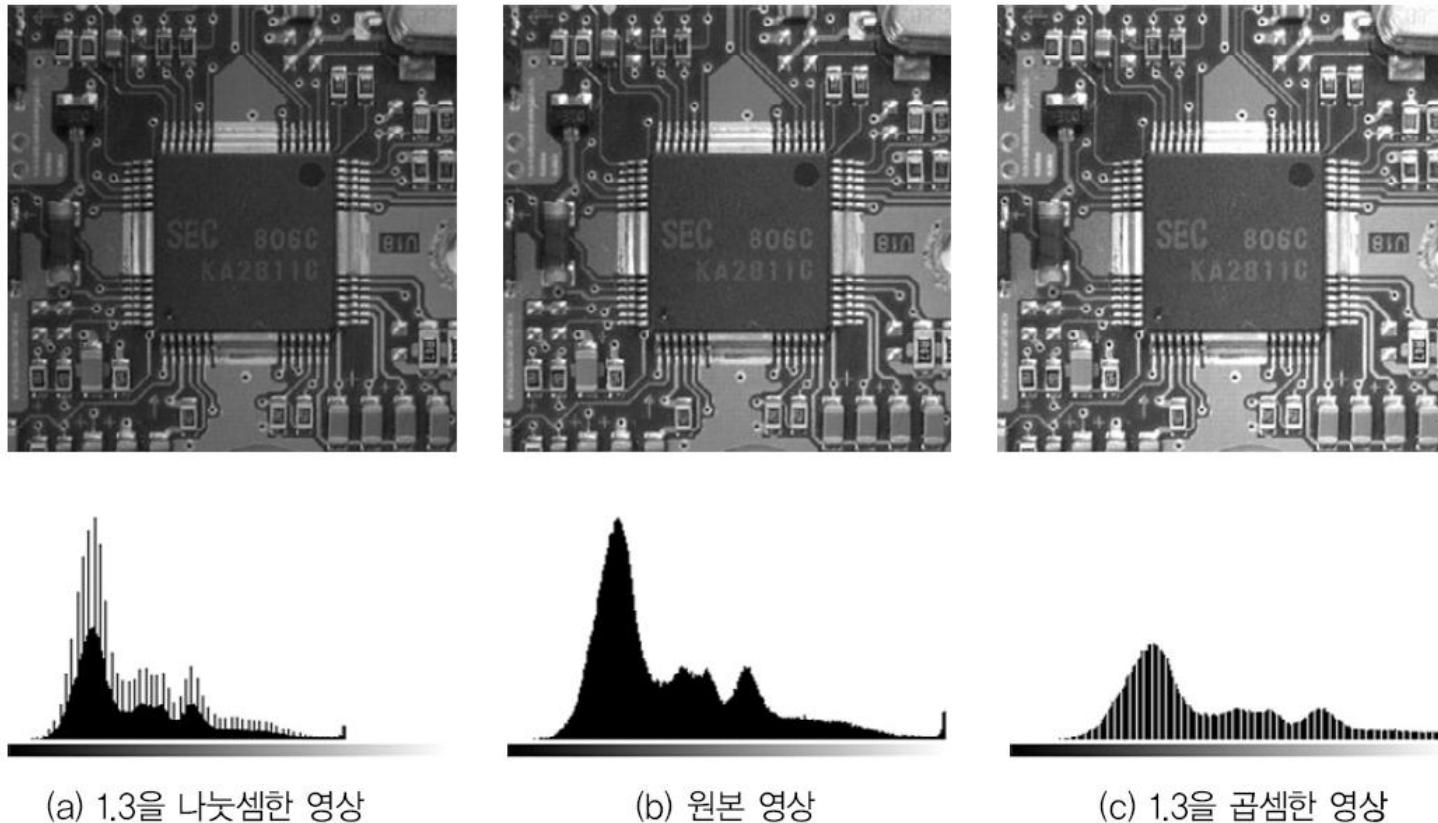
- 밝은 부분은 더욱 밝아지고, 어두운 부분은 약간 밝아져 영상 내의 밝기에 커다란 차이가 생기는 것
- 밝기의 차이가 커지므로 영상의 선명도 증가함.

## 화소 값의 나눗셈연산

- 화소 값을 임의의 상수 값으로 나누면 전체적으로 화소의 밝기 값은 감소하고, 최대 밝기와 최소 밝기의 차이는 작아짐.
- 밝은 부분은 많이 어두워지고, 어두운 부분은 약간 어두워짐.

화소 /  $\alpha$  : 영상의 밝기 차이 감소 = 희미해짐

## Section 02 산술 연산을 이용한 히스토그램 이동(계속)



[그림 5-5] 곱셈과 나눗셈연산으로 히스토그램 이동하기

- 곱셈연산: 명암 대비가 증가하여 히스토그램은 기둥의 분포 범위가 넓음.
- 나눗셈연산: 명암 대비가 감소하여 히스토그램의 분포 범위가 좁음.

## 디지털 영상의 산술연산[계속]



(a) 원본 영상



(b) 상수 1.3을 곱한 영상



(c) 상수 1.5를 곱한 영상



(d) 상수 1.7을 곱한 영상

[그림 4-7] 곱셈 상수 변화에 따른 디지털 영상의 명도 대비 향상

## 디지털 영상의 산술연산[계속]



(a) 원본 영상



(b) 상수 1.3으로 나눈 영상



(c) 상수 1.5로 나눈 영상



(d) 상수 1.7로 나눈 영상

[그림 4-8] 나눗셈 상수 변화에 따른 디지털 영상의 명도 대비 향상

# GRAYSCALE TRANSFORMATION

---

Improving image contrast and brightness  
by using **mapping function**

$$\mathbf{O}(x, y) = \mathbf{M}[\mathbf{I}(x, y)]$$

# SCALAR ARITHMETIC OPERATION

---

$$\mathbf{O}(x, y) = k \times \mathbf{I}(x, y) + l$$

$l$ : level,  $k$ : gain

## ❖ 클리핑(clipping) 처리

```
if( O(x, y) > 255 ) O(x, y) = 255;
```

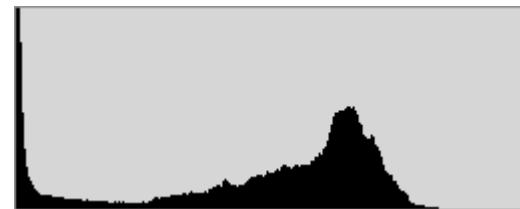
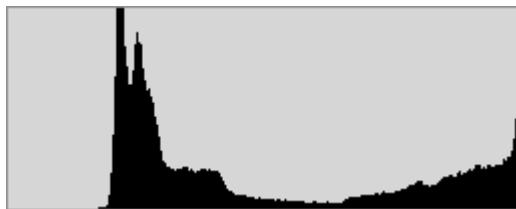
```
if( O(x, y) < 0 ) O(x, y) = 0;
```



$l = 50, k = 1$



$l = -50, k = 1$

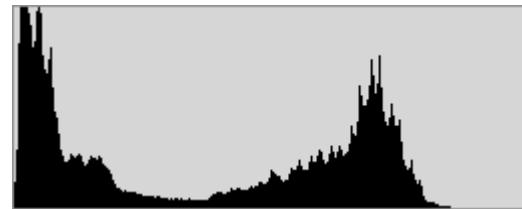




$l = 0, k = 1.2$



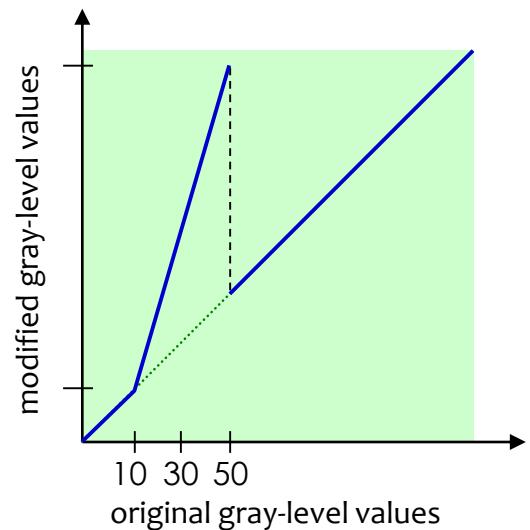
$l = 0, k = 0.83$

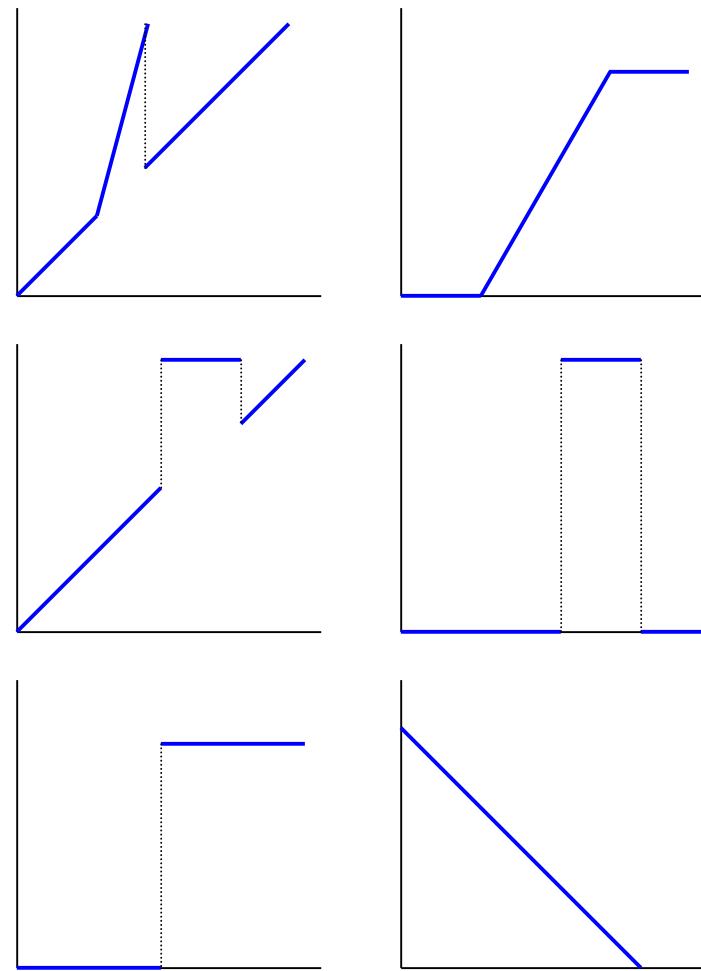
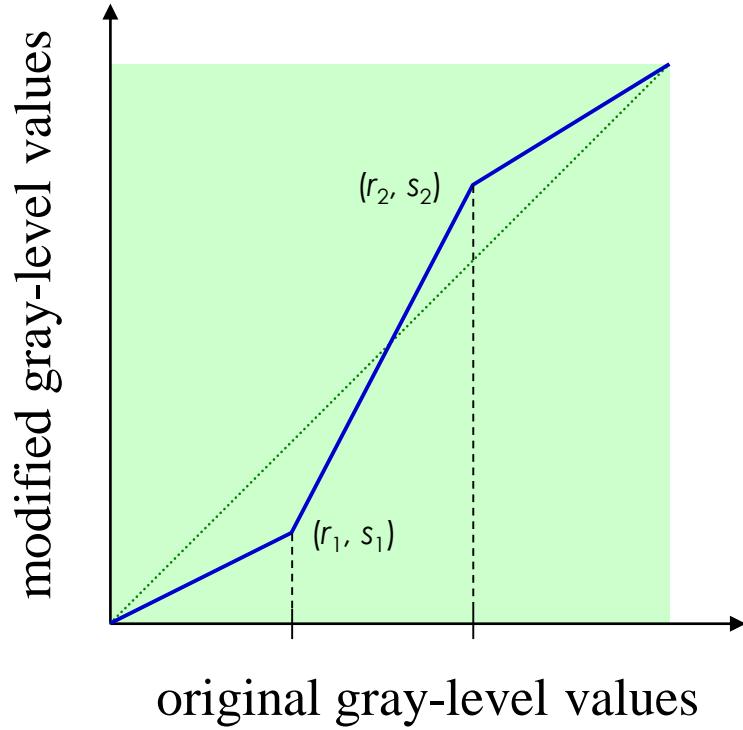


# example

(10,50) 범위의 gray level을 (10,250) 범위로 변경

$$\mathbf{M}[\mathbf{I}(x,y)] = \begin{cases} \mathbf{I}(x,y) & 0 \leq \mathbf{I}(x,y) < 10 \\ 6[\mathbf{I}(x,y)] - 50 & 10 \leq \mathbf{I}(x,y) \leq 50 \\ \mathbf{I}(x,y) & 50 < \mathbf{I}(x,y) \leq 255 \end{cases}$$

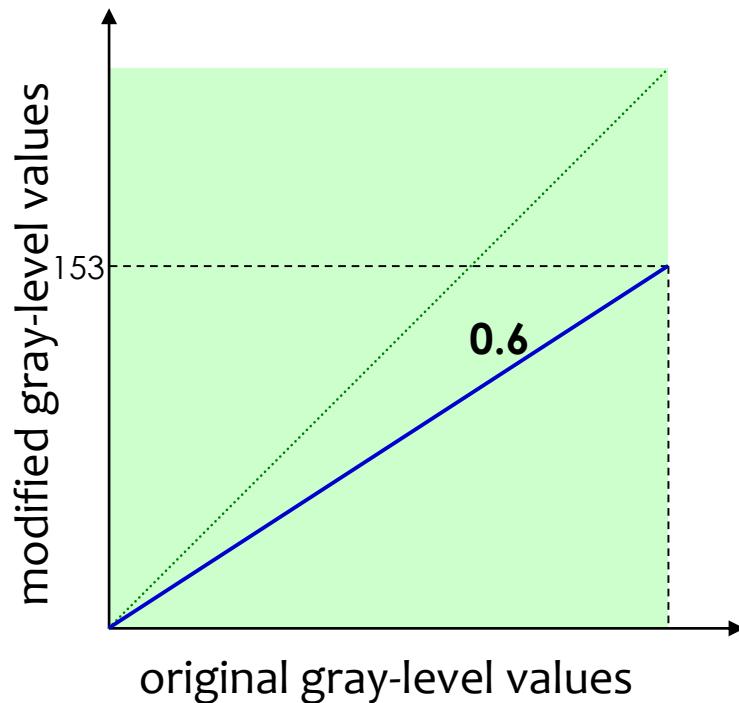




*General Form of Gray-Scale Modification*

# BRIGHTNESS SCALING BY MULTIPLICATION

## GRAYSCALE COMPRESSION

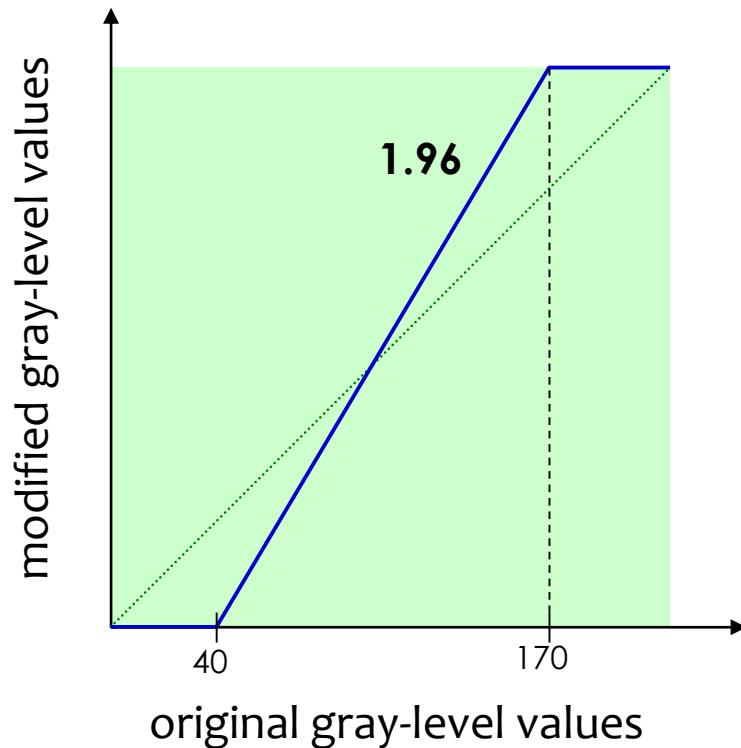


$$O(x, y) = 0.6[I(x, y)]$$

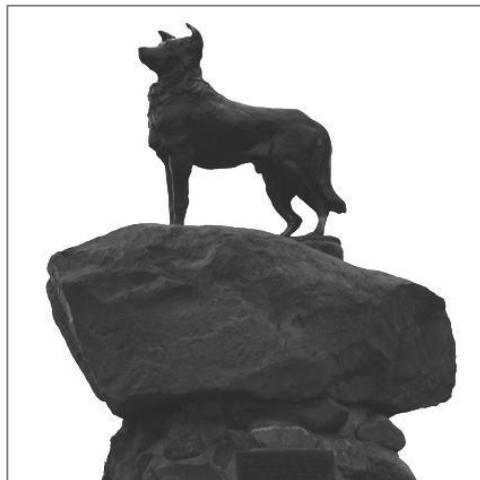


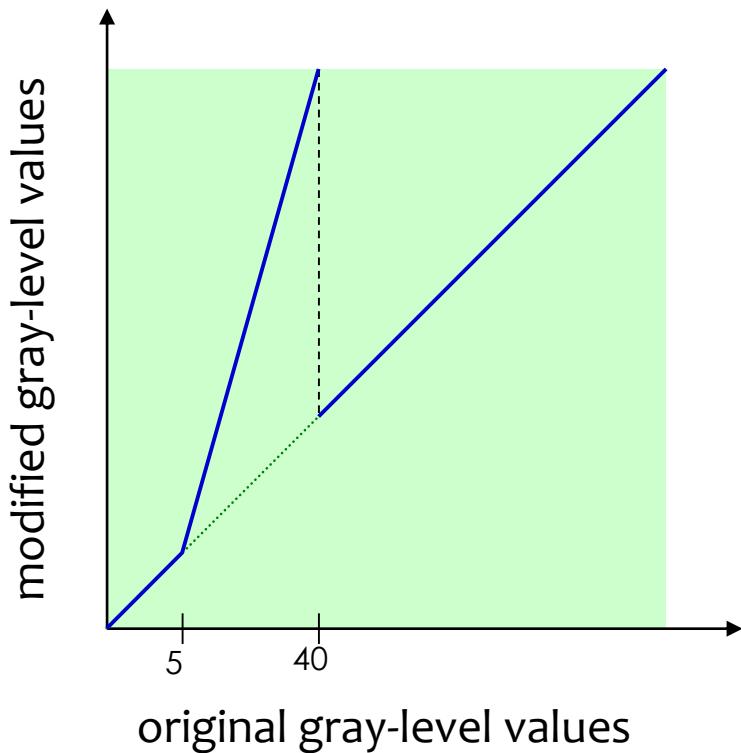
# BRIGHTNESS SCALING BY MULTIPLICATION

## GRAYSCALE STRETCHING



$$M[I(x,y)] = \begin{cases} 0 & 0 \leq I(x,y) < 40 \\ 1.96[I(x,y)] - 78.5 & 40 \leq I(x,y) \leq 170 \\ 255 & 170 < I(x,y) \leq 255 \end{cases}$$

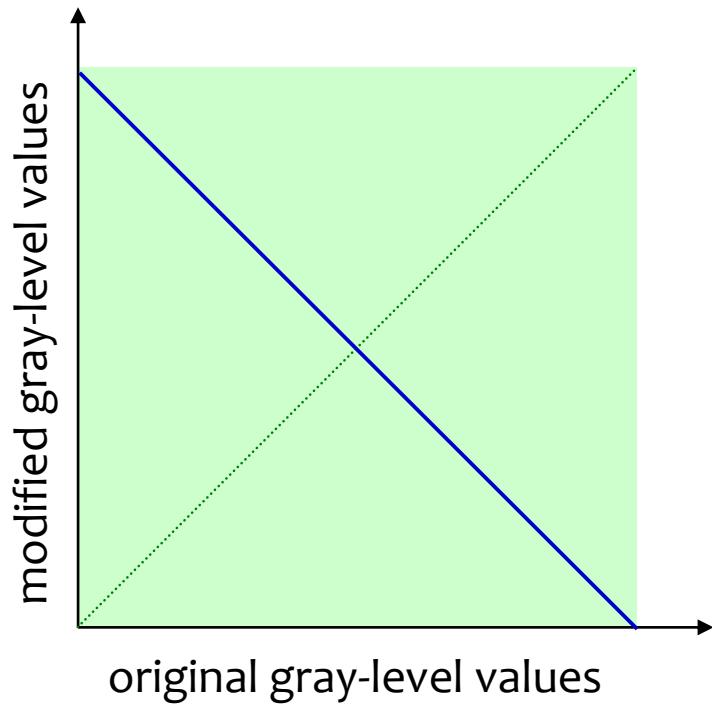




$$M[I(x,y)] = \begin{cases} I(x,y) & 0 \leq I(x,y) < 5 \\ 7.14[I(x,y)] - 30.7 & 5 \leq I(x,y) \leq 40 \\ I(x,y) & 40 < I(x,y) \leq 255 \end{cases}$$



# GRAY-LEVEL NEGATIVE



## ▶ 문제점

- 결과 값이 화소의 최대값과 최소값을 넘을 수 있음.

## ▶ 해결 방법

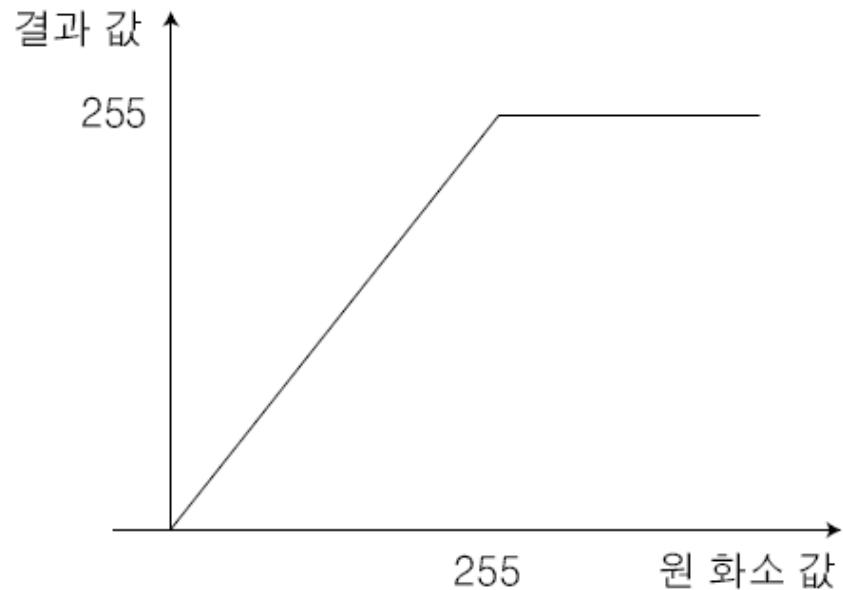
### ■ 클래핑(Clamping) 기법

- 연산의 결과 값이 최소값보다 작으면 그 결과 값을 최소값으로, 최대값보다 크면 결과 값을 최대값으로 하는 기법
- 8비트 그레이 영상의 최소값은 0, 최대값은 255
- 음수는 0으로 설정하고, 255보다 큰 값은 255로 설정함.

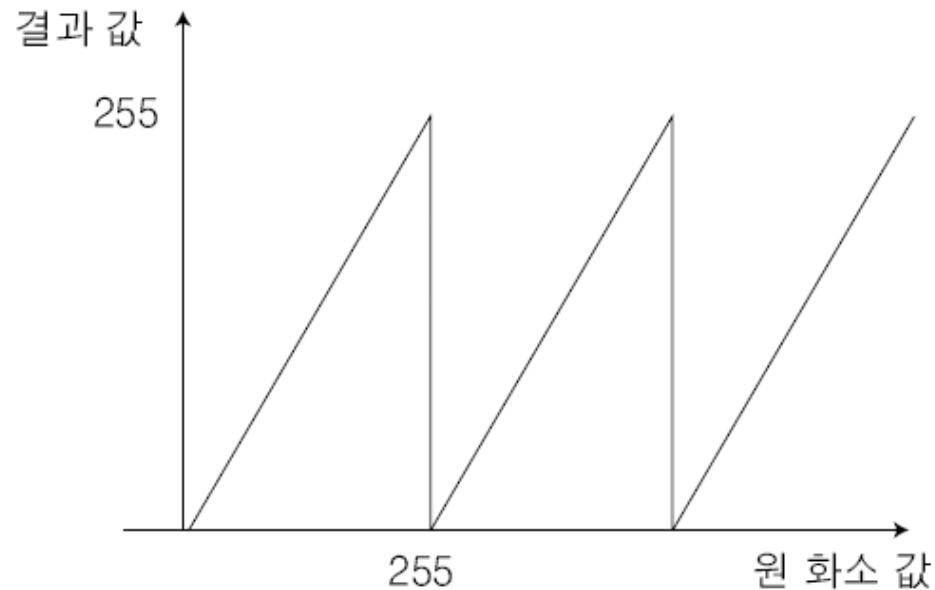
### ■ 랩핑(Wraping) 기법

- 연산의 결과 값이 최소값보다 작으면 그 결과 값을 최소값으로, 최대값보다 크면 최소값부터 최대값까지를 한 주기로 해서 이를 반복하는 기법
- 최대값+1은 최소값이 되고, 연산의 결과 값이 최대값+상수 값일 때는 계속 상수 값-1로 설정함.
- 8비트 그레이 영상의 최소값은 당연히 0이고, 최대값은 255
- 음수는 0으로, 255보다 큰 결과 값 256은 0으로, 257은 1로 설정한 후 이런 방식으로 주기를 계속 반복

## 산술연산의 문제점과 해결 방법(계속)



(a) 클래핑 기법



(b) 랩핑 기법

[그림 4-9] 산술연산의 문제점 해결 기법

# 학습 내용

---

- HISTOGRAM
- HISTOGRAM MODIFICATION

# HISTOGRAM MODIFICATIONS

---

Improving image contrast and brightness  
based on **histogram**

→ Focus on the histogram **shape** and **range**

## Section 03 히스토그램 스트레칭

### ▣ 히스토그램 스트레칭(Histogram Stretching)

- 명암 대비를 향상시키는 연산으로, 낮은 명암 대비를 보이는 영상의 화질을 향상시키는 방법
- 명암 대비 스트레칭이라고도 함.
- 히스토그램이 모든 범위의 화소 값을 포함하도록 히스토그램의 분포를 넓힐 힘.
- 기본 명암 대비 스트레칭과 앤드-인 탐색 기법이 대표적

## 기본 명암 대비 스트레칭

- ▣ 이상적이지 못한 히스토그램 분포 중에서 명암 대비가 낮은 디지털 영상의 품질을 향상시키는 기술
- ▣ 특정 부분이나 가운데에 집중된 히스토그램을 모든 영역으로 확장시켜서 디지털 영상이 모든 범위의 화소 값을 포함하게 함
- ▣ 기본 명암 대비 스트레칭 수행 공식

$$new\ pixel = \frac{old\ pixel - low}{high - low} \times 255$$

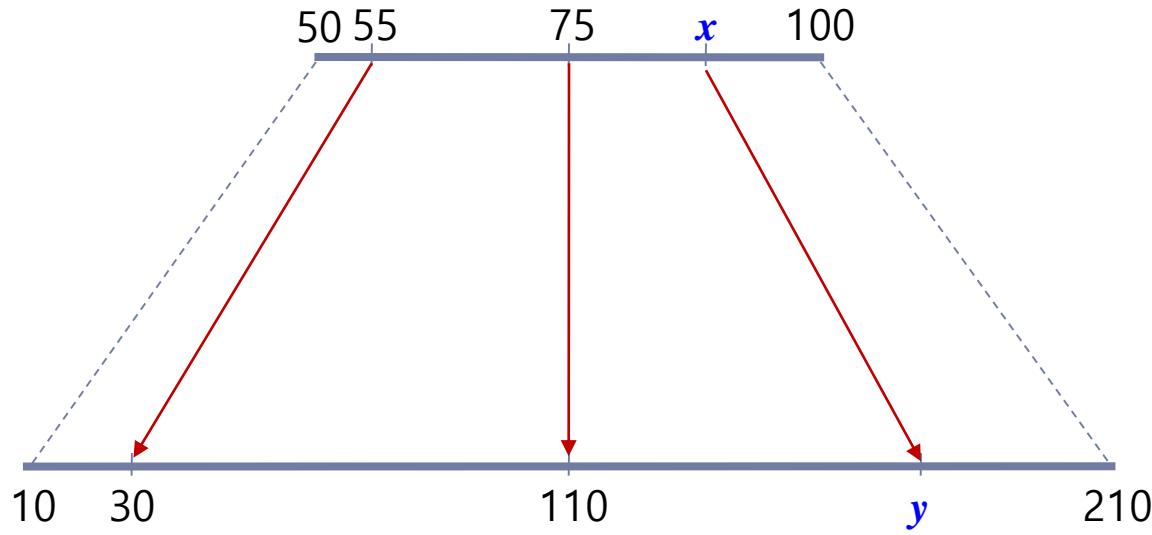
- old pixel은 원 영상 화소의 명도 값
- new pixel은 결과 영상 화소의 명도 값
- low는 히스토그램의 최저 명도 값
- high는 히스토그램의 최고 명도 값

## 앤드-인 탐색

- ▣ 일정한 양의 화소를 흰색이나 검정색으로 지정하여 히스토그램의 분포를 좀더 균일하게 만듦
- ▣ 앤드-인 탐색 수행 공식
  - 두 개의 임계 값(low, high) 사용

$$new\ pixel = \begin{cases} 0 & old\ pixel \leq low \\ \frac{old\ pixel - low}{high - low} \times 255 & low \leq old\ pixel \leq high \\ 255 & high \leq old\ pixel \end{cases}$$

# Scaling



$$(100 - 50) : (x - 50) = (210 - 10) : (y - 10)$$

$$(y - 10) * (100 - 50) = (x - 50) * (210 - 10)$$

$$y = \frac{(x - 50) * (210 - 10)}{(100 - 50)} + 10 = \frac{(210 - 10)}{(100 - 50)}(x - 50) + 10$$

$$I'(x, y) = \frac{(S_{\max} - S_{\min})}{(I_{\max} - I_{\min})}(I(x, y) - I_{\min}) + S_{\min}$$

$$\mathbf{O}(x, y) = \left[ \frac{S_{max} - S_{min}}{I_{max} - I_{min}} \right] [\mathbf{I}(x, y) - I_{min}] + S_{min}$$

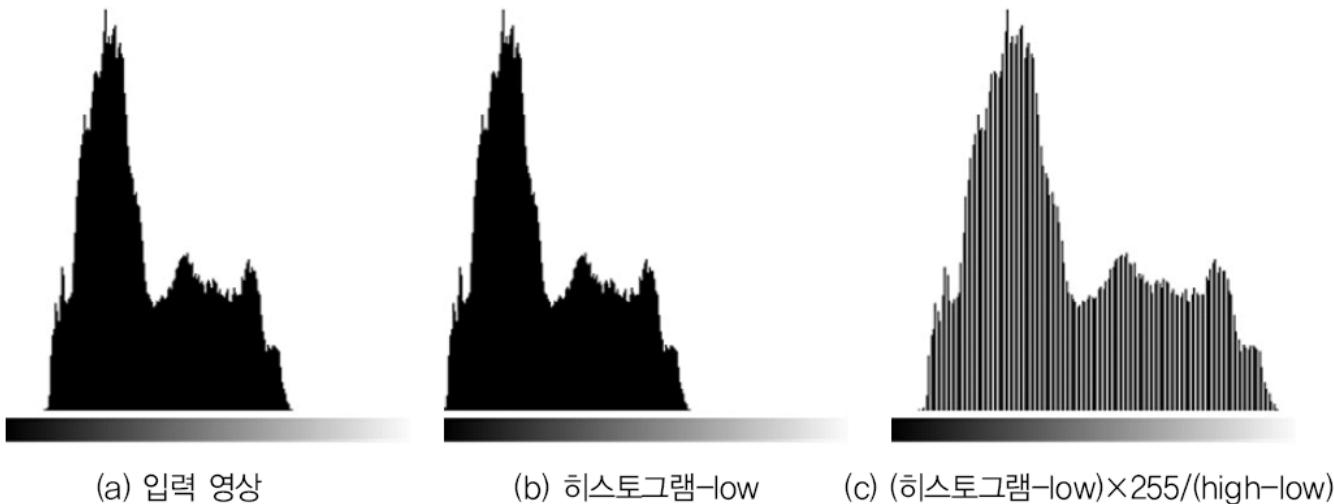
$I_{max}$  : largest gray-level value in the image  $\mathbf{I}(x, y)$

$I_{min}$  : smallest gray-level value in  $\mathbf{I}(x, y)$

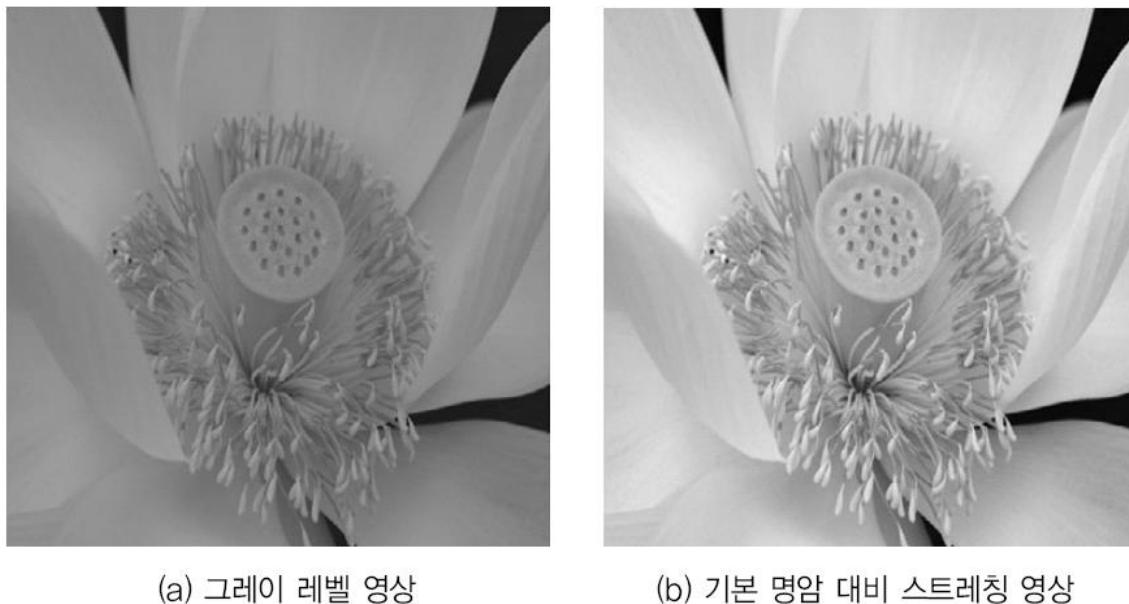
$S_{max}$  : maximum gray-level values possible

$S_{min}$  : minimum gray-level values possible

## 기본 명암 대비 스트레칭(계속)



[그림 5-7] 기본 명암 대비 스트레칭의 수행 과정



[그림 5-8] 영상에 기본 명암 대비 스트레칭을 적용한 결과 영상

## 엔드-인 탐색(계속)



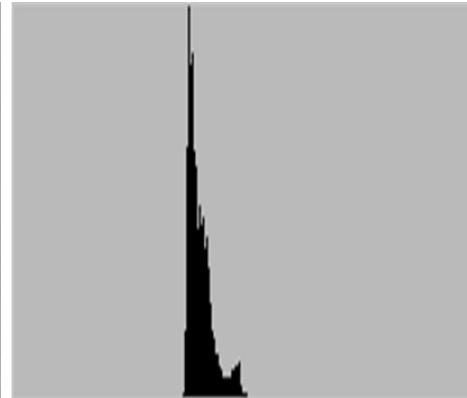
(a) 그레이 레벨 영상



(b) 엔드-인 탐색 영상

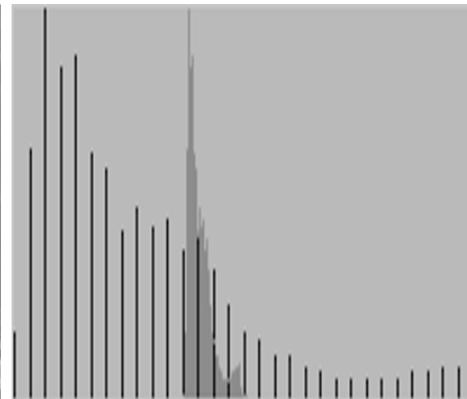
[그림 5-9] 엔드-인 탐색이 적용된 결과 영상

Low-contrast  
image

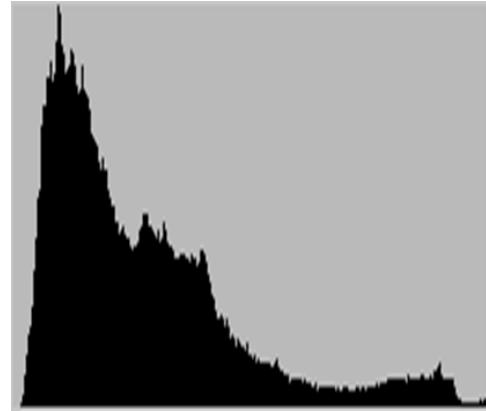


Histogram of  
low-contrast  
image

Image after  
histogram  
stretching



Histogram of  
image after  
stretching



Histogram  
of original  
image

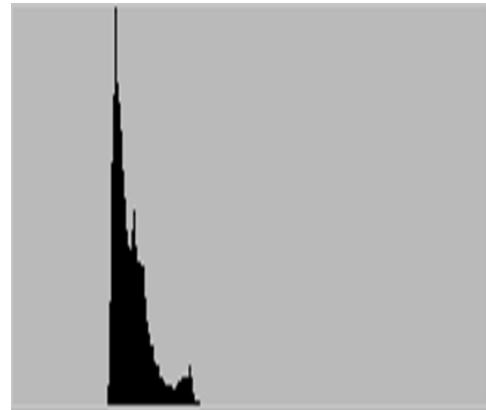


Image after  
shrinking

Histogram  
of shrunk  
image

# 히스토그램 평활화와 명세화

## ▶ 히스토그램 평활화 기술

- 편중된 디지털 영상의 히스토그램을 골고루 분산시켜 영상 전체의 명암 대비를 높여줌.

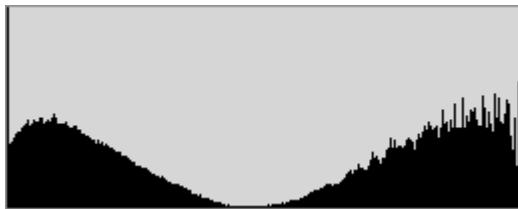
## ▶ 히스토그램 명세화 기술

- 디지털 영상이 원하는 히스토그램을 갖게 해주는 기술.
- 특정 부분의 명암 대비를 높일 수 있음.

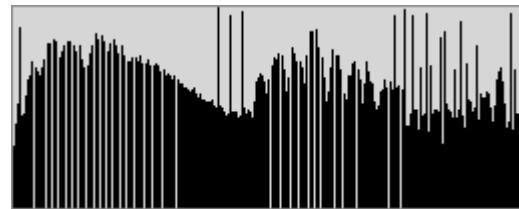


[그림 4-4] 디지털 영상의 명암 대비 변화

# Equalization



높은 contrast

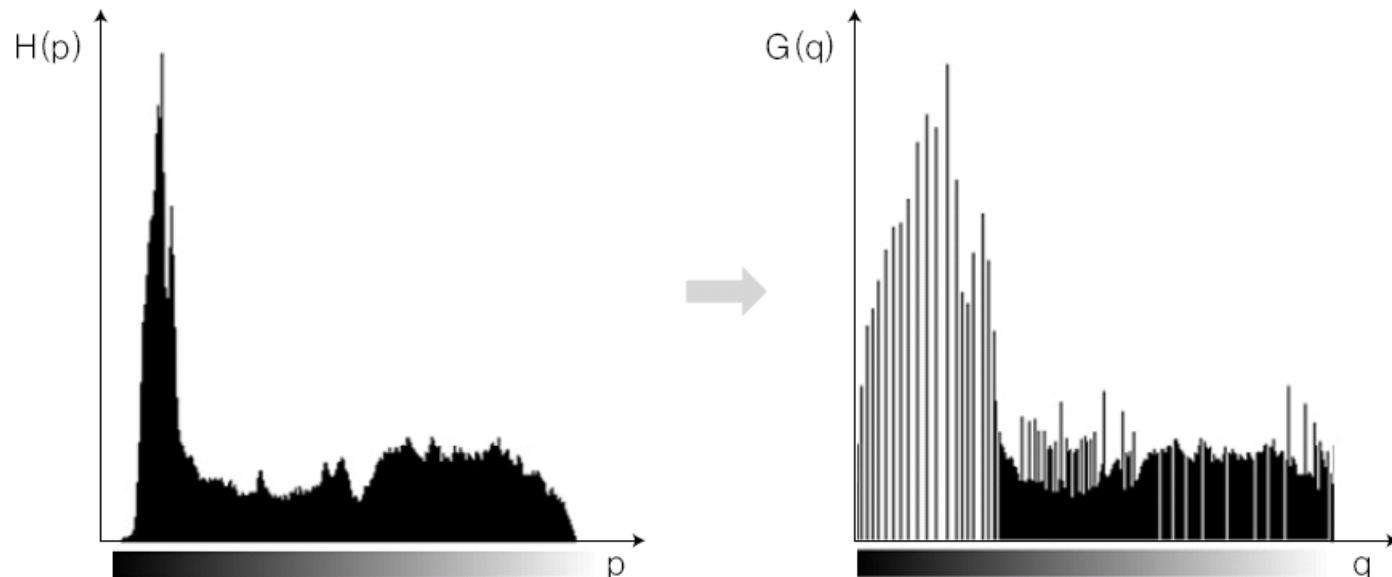


좋은 contrast

## Section 04. 히스토그램 평활화

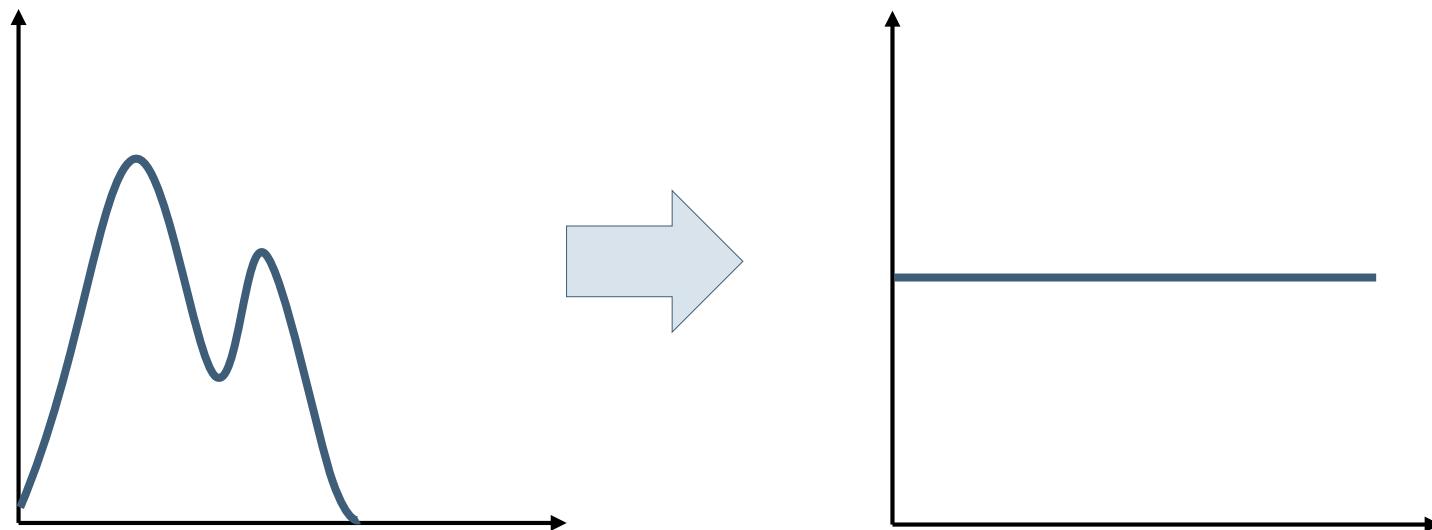
### ▣ 히스토그램 평활화 기법(Histogram Equalized)

- 어둡게 촬영된 영상의 히스토그램을 조절하여 명암 분포가 빠른 영상을 균일하게 만들어 줌.
- 영상의 밝기 분포를 재분배하여 명암 대비를 최대화
- 명암 대비 조정을 자동으로 수행
- 각 명암의 빈도는 변경하지 않음.
- 검출 특성이 좋은 영상만 출력하지는 않지만 영상의 검출 특성을 증가시킴

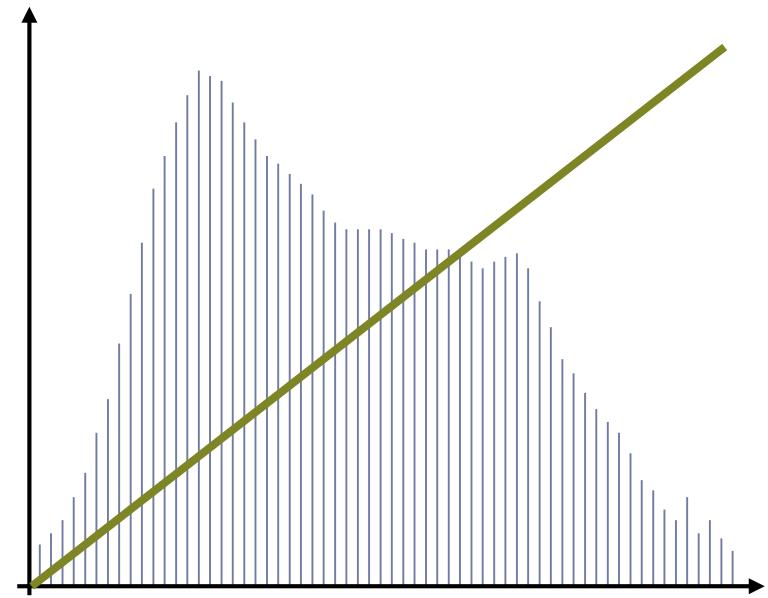
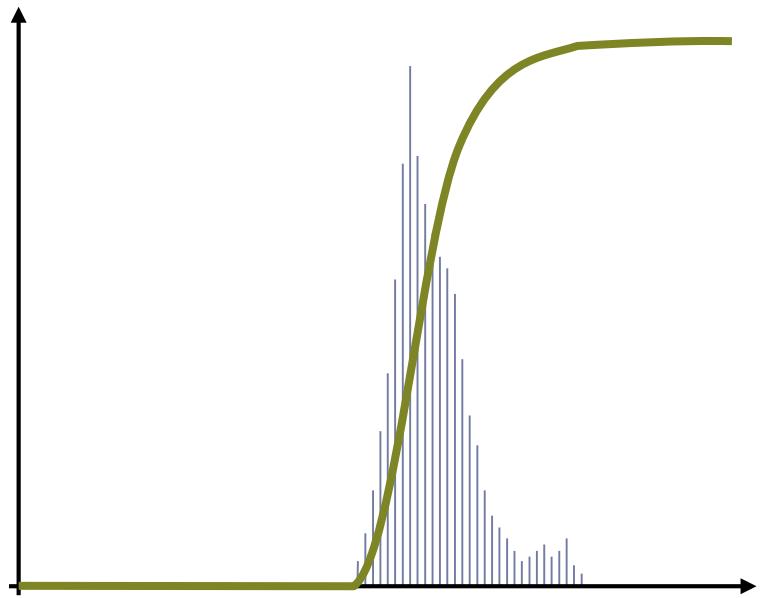


[그림 5-10] 히스토그램 평활화를 수행한 뒤 변화된 히스토그램 모습

Aim to change a picture in such a way as to produce a picture with *flatter* histogram, where all levels are equiprobable



## CDF Cumulative Distribution Function



# Deriving Algorithm (1)

$\mathbf{I}(l)$  and  $\mathbf{O}(l)$  ( $0 \leq l < L$ ):

for the input and output image, the number of pixels per level

$$\sum_{l=0}^{L-1} \mathbf{I}(l) = \sum_{l=0}^{L-1} \mathbf{O}(l)$$

for an arbitrarily chosen level  $p$  in the input image

$$\sum_{l=0}^p \mathbf{I}(l) = \sum_{l=0}^q \mathbf{O}(l)$$

## Deriving Algorithm (2)

Since the output histogram is uniformly flat  
( $T$ : total number of pixels in the image)

$$\mathbf{O}(l) = \frac{T}{N_{\max} - N_{\min}}$$

So the cumulative histogram of the output image

$$\sum_{l=0}^q \mathbf{O}(l) = q \times \frac{T}{N_{\max} - N_{\min}} = \sum_{l=0}^p \mathbf{I}(l)$$

# Deriving Algorithm (3)

Output pixels at level  $q$  is given by

$$E(q, \mathbf{I}) = q = \frac{N_{\max} - N_{\min}}{T} \times \sum_{l=0}^p \mathbf{I}(l)$$

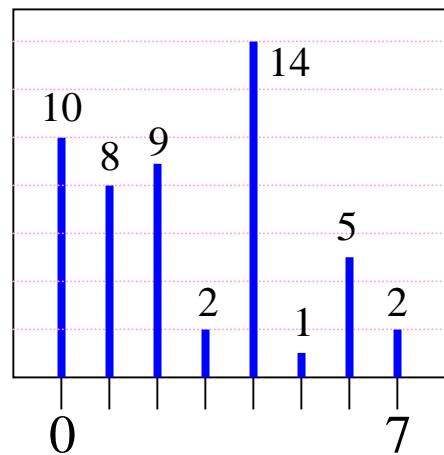
Equalizing function ( $E$ ) of the level ( $q$ ) and the image ( $I$ )

The output image is then

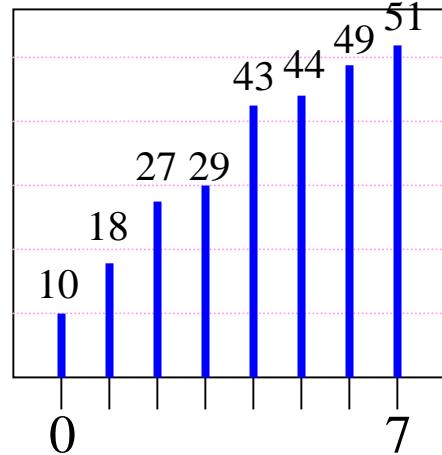
$$\mathbf{O}_{x,y} = E(\mathbf{I}_{x,y}, \mathbf{I})$$

# algorithm

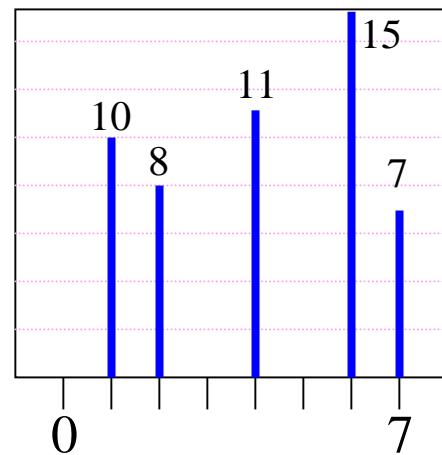
- ① 입력 영상의 히스토그램의 값을 누적시켜 각 레벨에서의 히스토그램 누적 합 계산
- ② 히스토그램의 누적 합을 전체 픽셀의 개수로 나누어 값을 정규화함
- ③ 정규화된 값에 최대 gray level 값을 곱한 후 반올림을 수행
- ④ 입력 영상의 각 gray level에 대해 변환 값으로 대응



히스토그램



누적값



균일화 결과

$$\frac{(10, 18, 27, 29, 43, 44, 49, 51)}{51} \times 7$$

$$\approx (1.37, 2.47, 3.71, 3.98, 5.90, 6.04, 6.73, 7.00)$$

$$\approx (1, 2, 4, 4, 6, 6, 7, 7)$$

# LUT

4	5	3	6	7
4	2	2	4	6
0	2	2	5	7
0	0	2	3	5
0	1	2	4	4

입력레벨	개수	누적값	균일화	결과레벨
0	10	10	1.37	1
1	8	18	2.47	2
2	9	27	3.71	4
3	2	29	3.98	4
4	14	43	5.90	6
5	1	44	6.04	6
6	5	49	6.73	7
7	2	51	7.00	7

6	6	4	7	7
6	4	4	6	7
1	4	4	6	7
1	1	4	4	6
1	2	4	6	6

# 히스토그램 평활화의 3단계

## ① 1단계

- 명암 값  $j$ 의 빈도 수  $hist[j]$ 를 계산해 입력 영상의 히스토그램 생성

## ② 2단계

- 각 명암 값  $i$ 에서 0~ $i$ 까지의 누적 빈도 수(누적합)를 계산

$$sum[i] = \sum_{j=0}^i hist[j]$$

## ③ 3단계

- 2단계에서 구한 누적 빈도 수를 정규화(정규화 누적합)

$$n[i] = sum[i] \times \frac{1}{N} \times I_{\max}$$

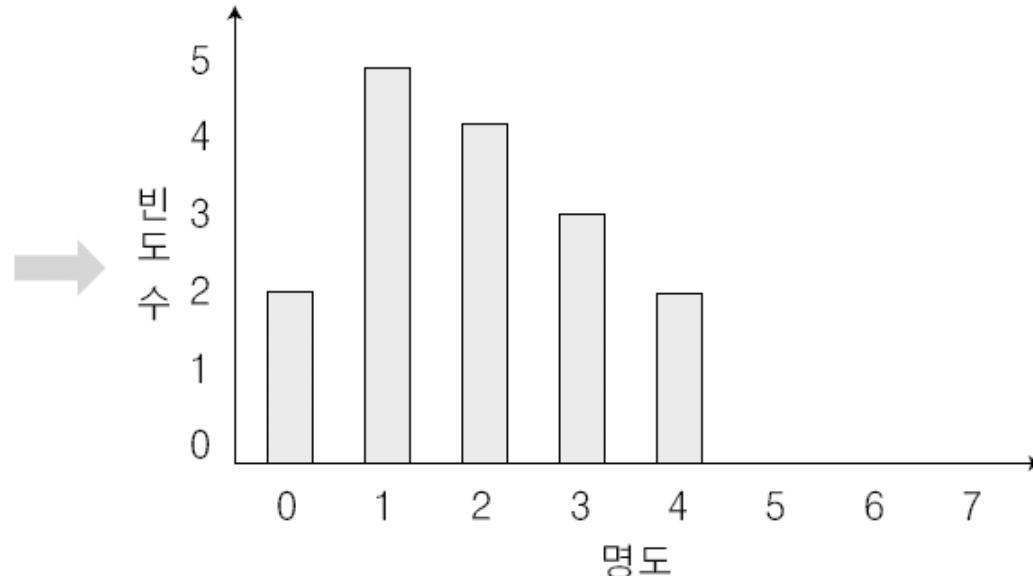
- $N$ 은 화소의 총 수,  $I_{\max}$ 는 최대 명도 값
- 3단계에서 얻은 정규화된 값  $n[i]$ 로 입력 영상의 화소 값  $i$ 를 변환하면 평활화된 결과 영상 생성

# 히스토그램 평활화\_1단계

## 1단계

- 빈도 수  $hist[j]$ 에서의 히스토그램 생성

2	4	4	3
2	1	3	3
1	0	1	2
0	1	1	2



(a) 입력 영상

(b) 히스토그램

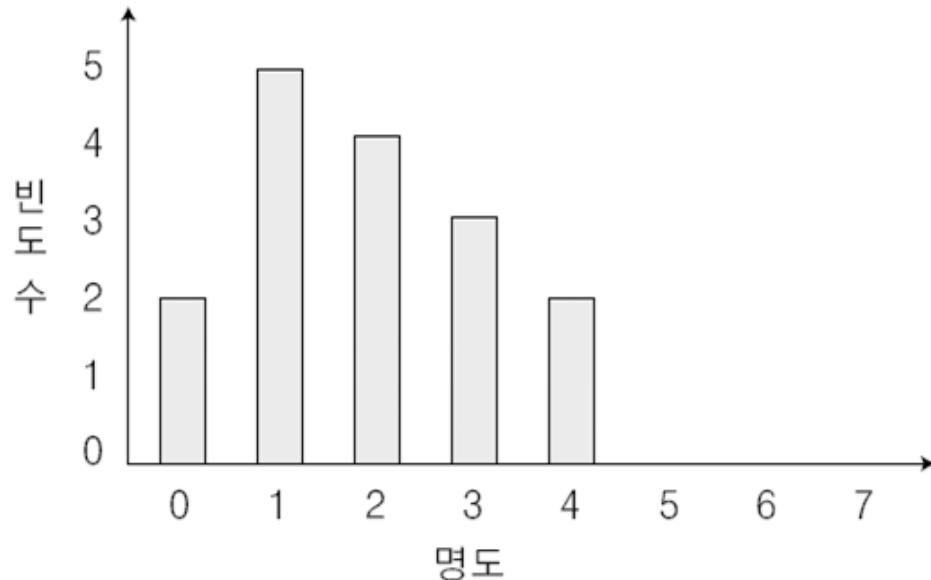
[그림 5-11] 히스토그램의 생성

- 화소의 명도 값 0은 2개, 1은 5개, 2는 4개, 3은 3개, 4는 2개
- 가장 큰 명도 값이 40이므로 전체적으로 왼쪽으로 치우침.

# 히스토그램 평활화\_2단계

## 2단계

- 누적합 sum[i] 생성



(a) 히스토그램

명도	누적합
0	2
1	7
2	11
3	14
4	16
5	16
6	16
7	16

(b) 누적합

[그림 5-12] 히스토그램에서 누적합 계산

- 화소의 명도 0번까지의 누적합은 2, 1번까지는  $2+5=7$ , 2번까지는  $2+5+4=11$ , 3번까지는  $2+5+4+3=14$ , 4번까지는  $2+5+4+3+2=16$
- 나머지 명도 값은 영상에는 없으므로 누적합은 16

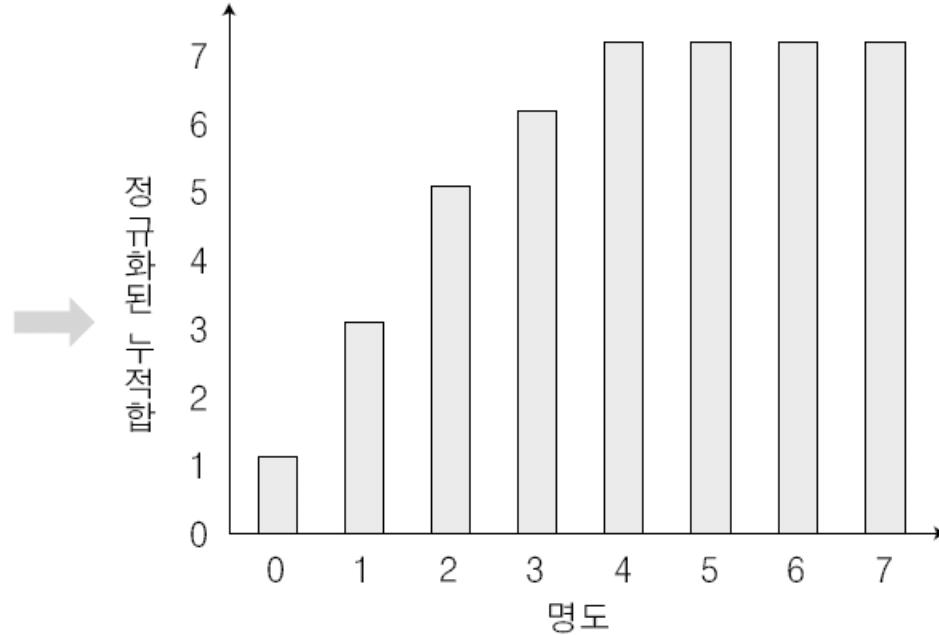
# 히스토그램 평활화\_3단계

## 谤 3단계

■  $n[i] = \text{sum}[i] * (1/16) * 7$

명도 (i)	누적합 (sum [i])	정규화된 누적합 (n [i])
0	2	0.875
1	7	3.0625
2	11	4.8125
3	14	6.125
4	16	7
5	16	7
6	16	7
7	16	7

(a) 정규화

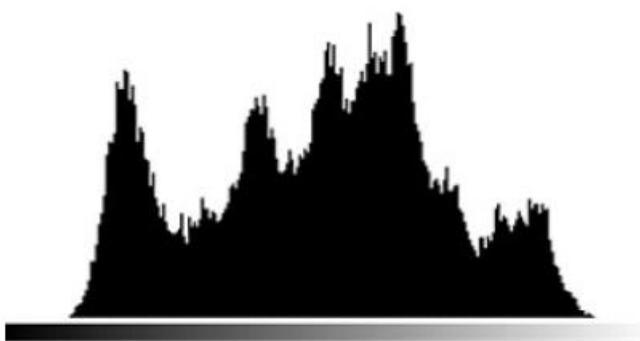


(b) 정규화된 히스토그램

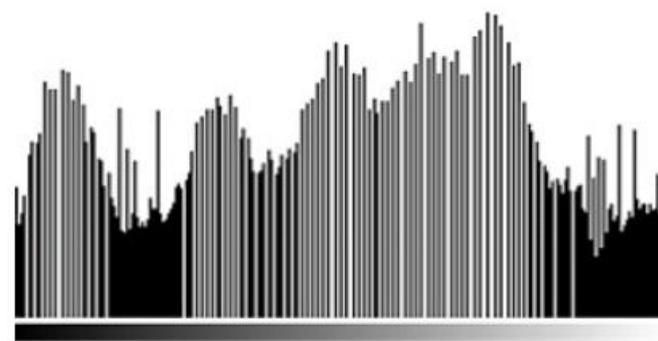
[그림 5-13] 누적합에서 정규화

- $n[0]$ 은  $2 * (1/16) * 7 = 0.875$ ,  $n[1]$ 은  $7 * (1/16) * 7 = 3.0625$
- $n[2]$ 는  $11 * (1/16) * 7 = 4.8125$ ,  $n[3]$ 은  $14 * (1/16) * 7 = 6.125$
- $n[4]$ 와  $n[5]$ ,  $n[6]$ ,  $n[7]$ 은  $16 * (1/16) * 7 = 7$

## 히스토그램 평활화를 적용한 영상



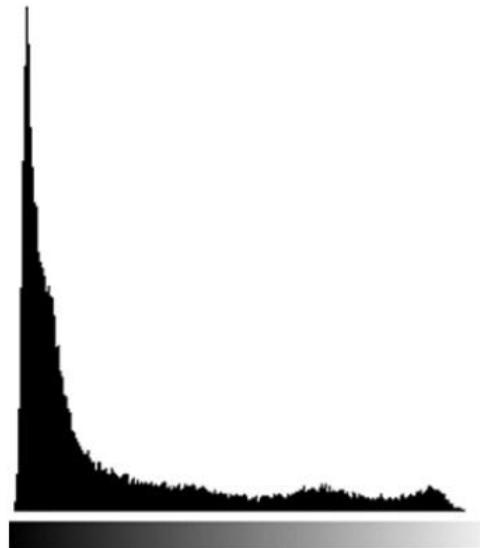
(a) 원본 영상



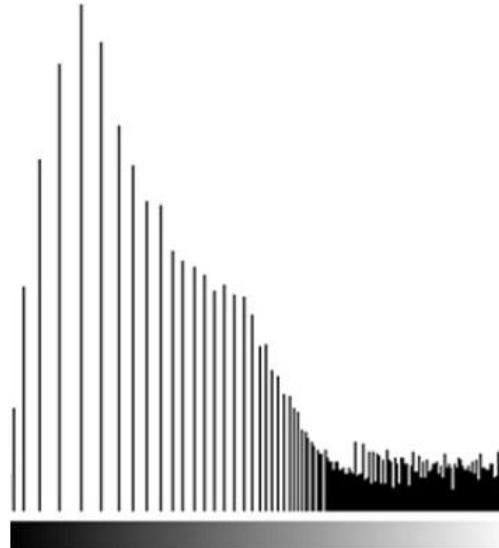
(b) 평활화 영상

[그림 5-15] 레나(Lenna) 영상의 평활화와 평활화한 영상의 히스토그램

## 히스토그램 평활화를 적용한 영상(계속)

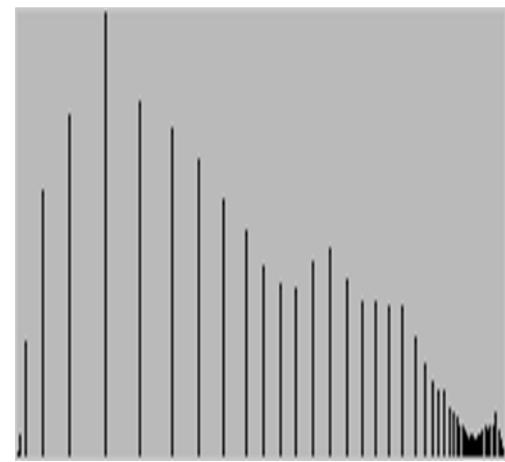
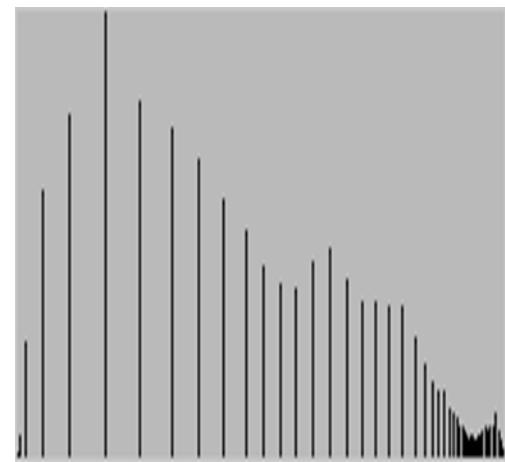


(a) 원본 영상

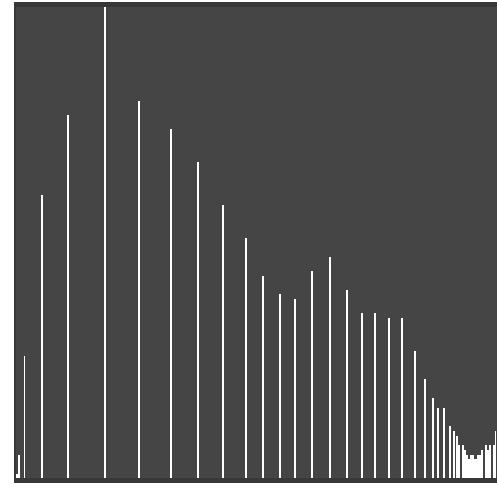


(b) 평활화 영상

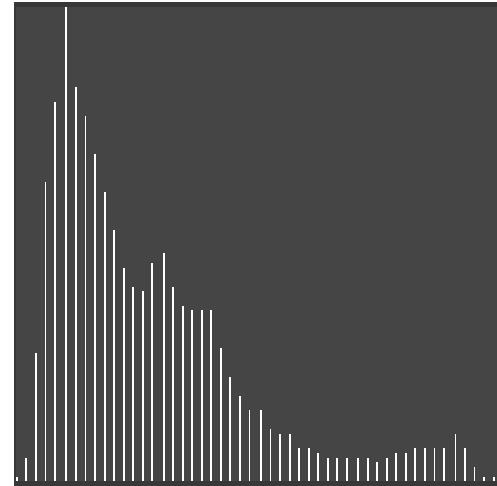
[그림 5-16] 명암 대비가 낮은 영상의 평활화와 평활화한 영상의 히스토그램



# Histogram normalization vs. equalization



equalization

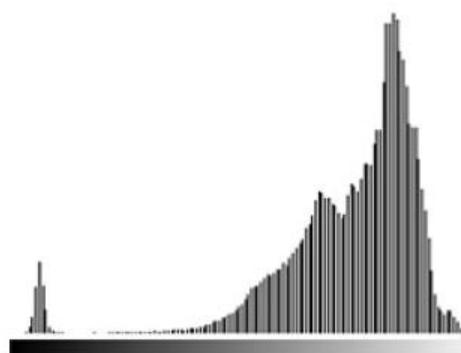
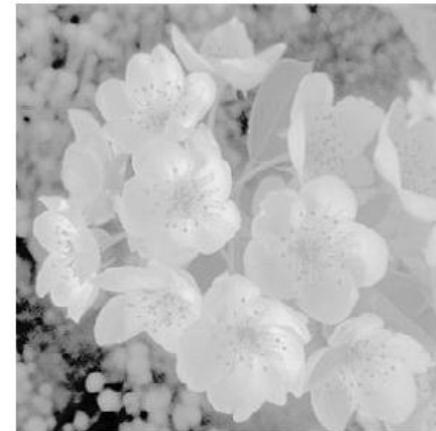


normalization

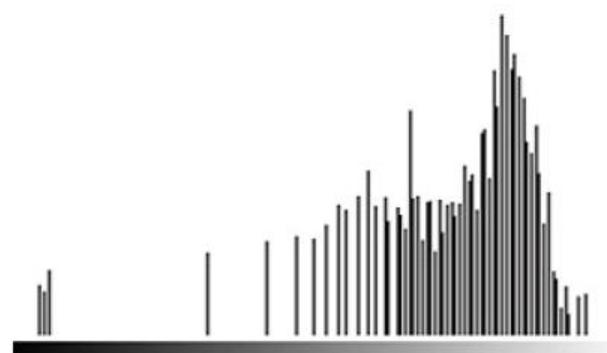
## 히스토그램 명세화 개념



(a) 원본 영상과 히스토그램



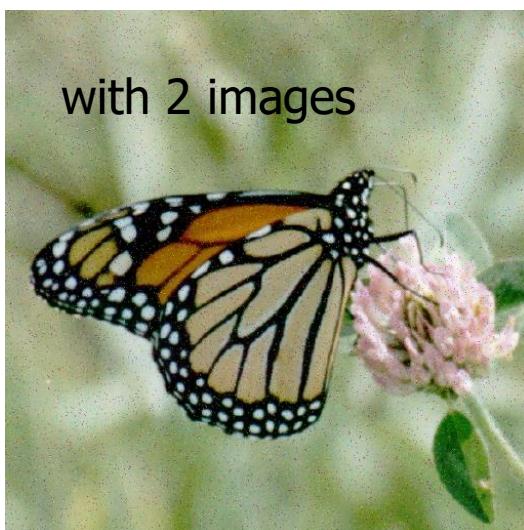
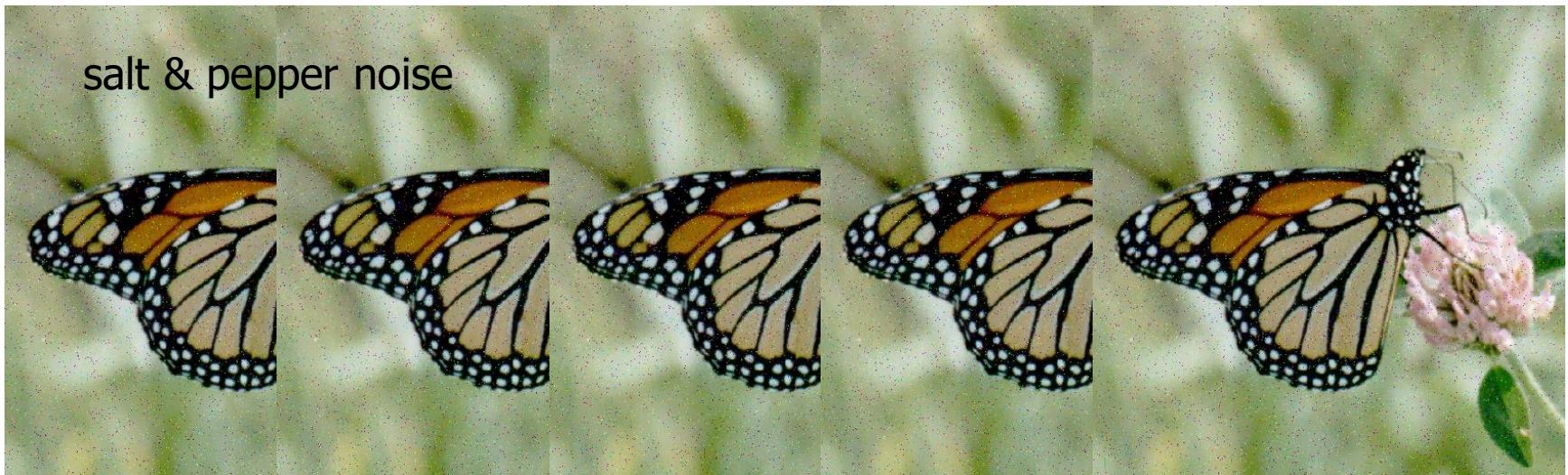
(b) 원하는 히스토그램



(c) 명세화된 영상과 히스토그램

[그림 5-17] 히스토그램의 명세화 개념

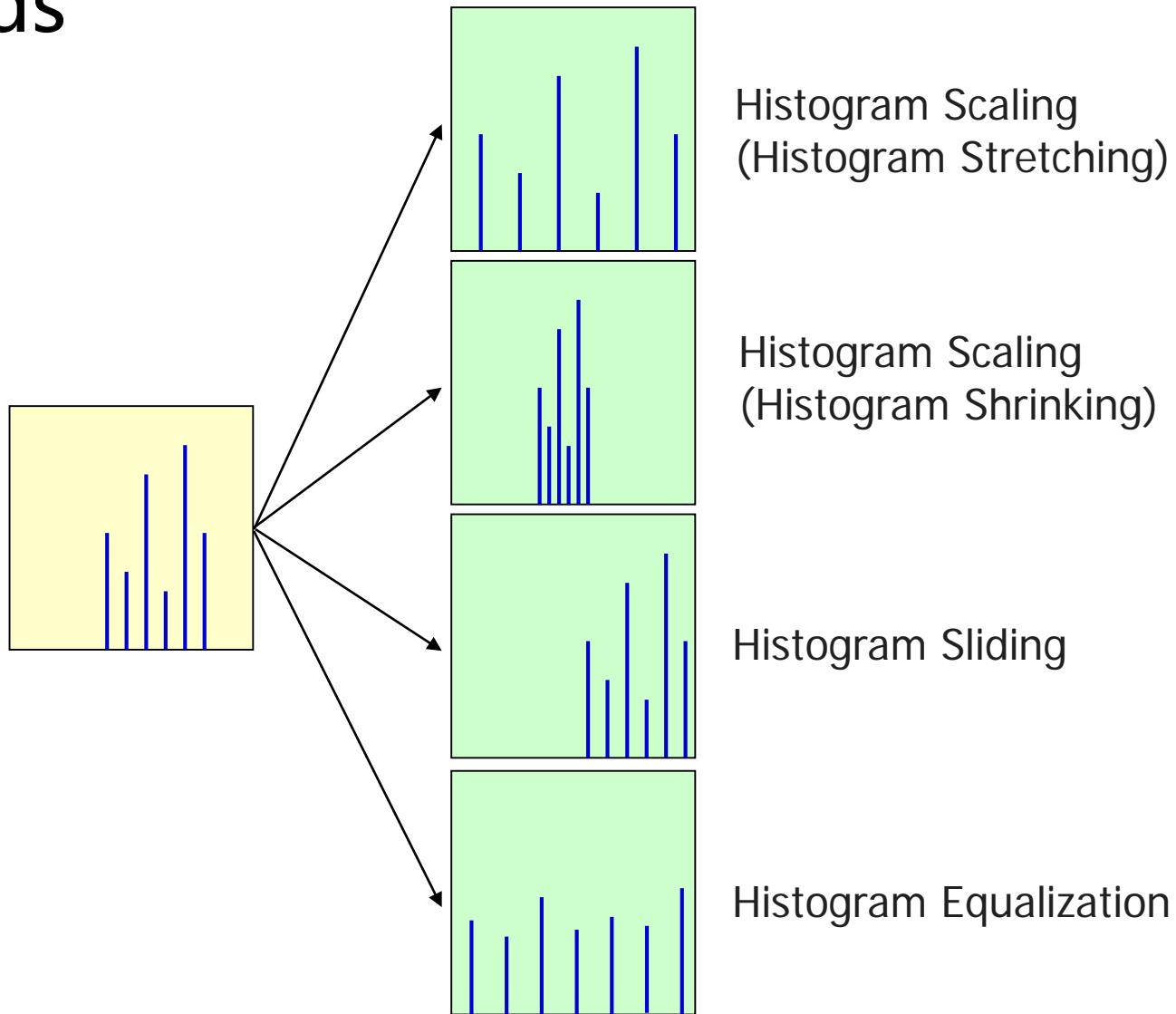
# IMAGE ARITHMETIC OPERATION



averaging

영상은 CVIPTools로 부터 가져옴

# fields



# DISADVANTAGE

---

- Background noise can be increased
- The image quality in a near-constant region may be degraded

# REFERENCE

---

- R. Gonzalez, R. Woods, **Digital Image Processing (2nd Edition)**, Prentice Hall, 2002
- Scott E Umbaugh, **Computer Imaging**, CRC Press, 2005
- Mark Nixon and Alberto Aguado, **Feature Extraction & Image Processing**, ELSEVIER, 2008
- Frank SHIH, **Image Processing and Pattern Recognition**, IEEE Press, 2010



Thank you